8030

CONTROL DATA® 8030
RECORD TRANSMISSION SYSTEM
USER'S MANUAL

Any comments concerning this manual should be addressed to:

CONTROL DATA CORPORATION
Publication Department
9549 PENN AVENUE SOUTH
MINNEAPOLIS, MINNESOTA 55431

June, 1965
Pub. No. 368 229 00

1965, Control Data Corporation
Printed in the United States of America

CONTENTS

INTRODUCTION v

GENERAL DESCRIPTION 1

INPUT FILES 3
- A. Magnetic Tape 3
- B. Punched Cards 4

OUTPUT FILES 4
- A. Magnetic Tape 4
- B. Line Printer 5

DATA TRANSMISSION 6
- A. Data Channel 6
- B. Data Flow 6
- C. Processing Overlap 7
- D. Error Protection 7

LOCAL PERIPHERAL PROCESSING 9

TERMINAL OPERATION 9
- A. Selection 9
- B. Control 11
- C. Notification 12
- D. Job Processing 12

APPENDICES
- A. Throughput Calculation 13
- B. STEP Alteration 19
- C. BCD Codes 31
- D. Minimum Terminal Configuration 33

ILLUSTRATIONS

Figure 1. 8030 Record Transmission System vi

Figure 2. 8030 Information Flow 2

Figure 3. Basic Record Cycle 8

Figure 4. Record Transmission Control Panel 10

Figure 5. Timing Relationships Between Processor Operations During Record Transmission 18

Figure 6. 8092-B Storage Layout 20

Figure 7. Sample Input Subprogram 29

Figure 8. Sample Output Subprogram 30

8030 RECORD TRANSMISSION SYSTEM

INTRODUCTION

This document describes the performance characteristics of the **CONTROL DATA®** 8030 Record Transmission System (figure 1) which uses the CONTROL DATA 8092 Tele-Programmer (a stored program processor) as the major unit and transmits over standard voice bandwidth telephone circuits. The system includes a Standard TErminal Program (STEP) to support the processor and is available as a complete package installed and operating in any area where reliable, medium speed, point-to-point data transmission is required. A STEP option permits system terminals to be used for local peripheral operations, such as data medium conversion, whenever they are not engaged in data transmission.

Advanced versions of the 8030 System which use the full range of available communications channels, including microwave transmission and coaxial cable connection, can also be provided but are not described in this manual.

Figure 1. 8030 Record Transmission System

GENERAL DESCRIPTION

The CONTROL DATA 8030 Record Transmission System is a complete hardware/software package which performs point-to-point data transmission and local peripheral processing. The system consists of two terminal facilities interconnected by a voice bandwidth telephone circuit and associated modulation/demodulation equipment (modems).

Each terminal facility is made up of the following functional units:

Processor and Control Panel

This unit stores and checks data and controls the terminal. It consists of a CONTROL DATA 8092-B TeleProgrammer and the 8095 Record Transmission Control Panel.

The Data Set Controller

This unit interfaces the 8030 Processor to a standard DATA-PHONE® Data Set 201 (or equivalent modem) connected to a full- or half-duplex voice-grade circuit. Its major function is serial/parallel data conversion.

Peripheral Equipment

This equipment consists of devices and controllers for the appropriate input or output functions. Any Control Data peripheral equipment which connects to the 8092-B is usable with the 8030 System if the input or output subprogram to drive that particular equipment is available.

The 8030 Standard Terminal Program (STEP)

This program provides complete communications and operator control for the terminal for both transmission and peripheral processing operations. STEP also provides several specific input/output subprograms which drive the following peripheral equipment:

1. A CONTROL DATA 601 Magnetic Tape Unit
2. A CONTROL DATA 166-2 Printer
3. A CONTROL DATA 405 Card Reader
4. A CONTROL DATA 603 Magnetic Tape Unit

DATA-PHONE® is a registered trademark of Bell System.

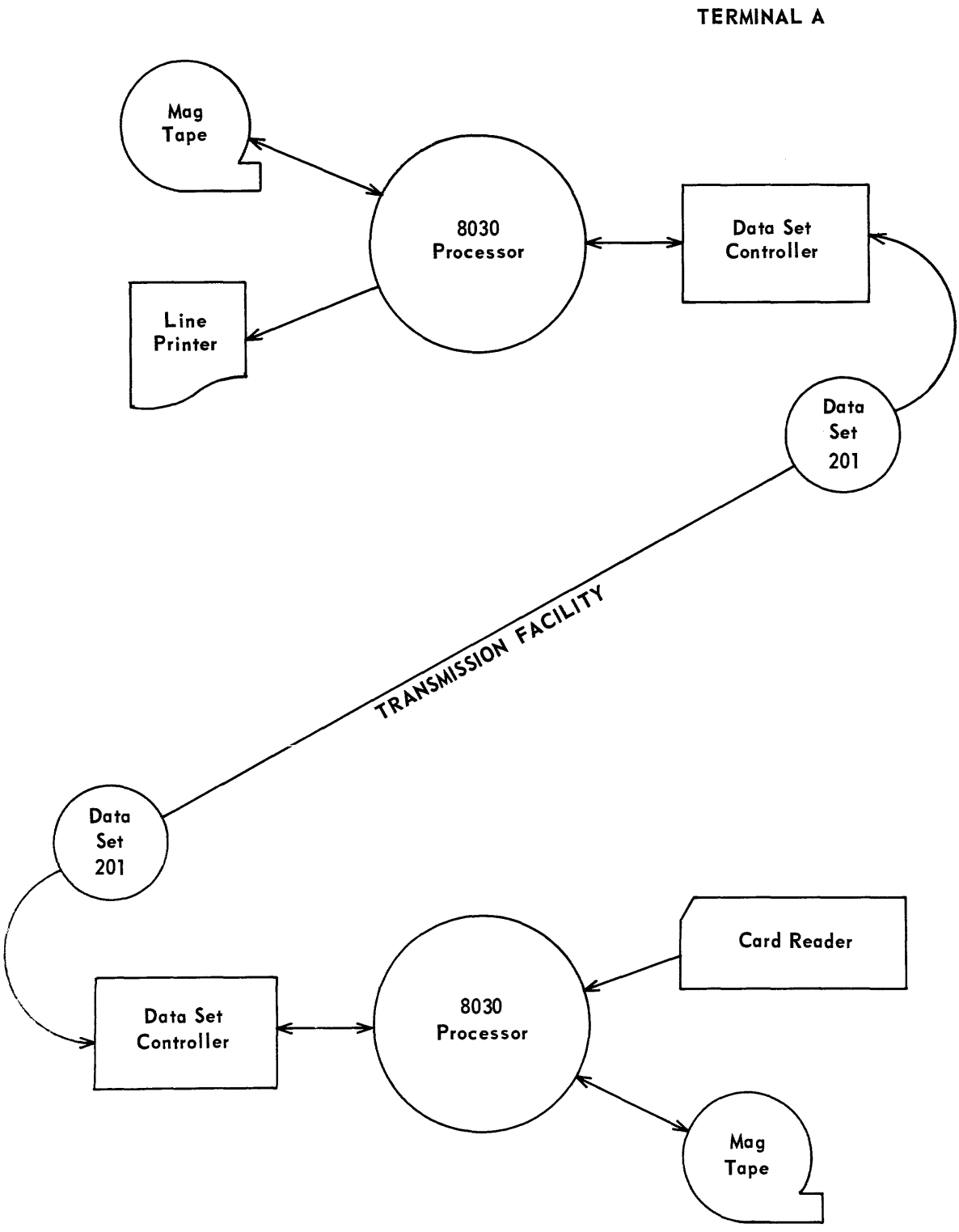


Figure 2. 8030 Information Flow

Using appropriate combinations of these peripherals, STEP can perform:

1. Local card to magnetic tape
2. Local card to printer
3. Local magnetic tape to printer
4. Local magnetic tape to magnetic tape
5. Local card to remote magnetic tape
6. Local card to remote printer
7. Local magnetic tape to remote magnetic tape
8. Local magnetic tape to remote printer

The specific information formats and codes for these operations are described in this manual.

In addition to this complete processing facility, the 8030 System through STEP allows each user to incorporate his own specific input/output subprograms for special editing functions or to accommodate peripheral devices (such as paper tape readers, paper tape punches and teletypewriters) not previously listed.

INPUT FILES

Standard STEP input subprograms provide the ability to process magnetic tape or punched card files.

A. Magnetic Tape

A magnetic tape file is that collection of physical tape records between input tape position at start of job and the next end-of-file record in a forward direction.

Records comprising a file can have odd or even parity and can vary in length. Maximum permissable length is 1000 characters. Recording density can be 200 or 556 characters per inch. A given file can contain a mixture of odd and even parity records but must be of uniform density.

By considering beginning-of-file an implicit function of tape position, STEP can process multi-file tapes as well as single-file tapes. Multi-tape files can be processed if an end-of-file record terminates each reel. Trailer labels can be processed as a one-record file if they are followed by an additional end-of-file record.

B. Punched Cards

A punched card file is a collection of 80-column cards followed by a special end-of-file card punched "XXX" in card columns 1, 2, and 3. "XXX" must be legitimate Hollerith characters and can be specified by the user. The cards can be binary coded (indicated by a 7 and 9 punch in column 1) or Hollerith coded. Both binary cards and Hollerith cards can be included in the same file.

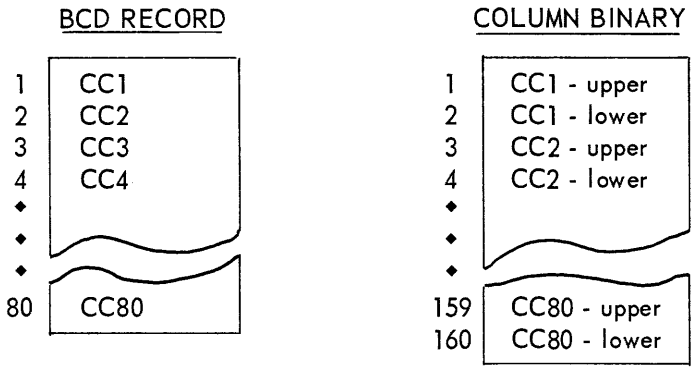
OUTPUT FILES

An output file results from the processing of an input file. The characteristics of the output file depend upon output and input mediums. Standard output subprograms when used with standard input subprograms produce files with the following characteristics:

A. Magnetic Tape

Input from Magnetic Tape. An output file resulting from tape input normally is an exact record-by-record duplicate of the input file (including the end-of-file record). If a non-recoverable tape error occurs, operator intervention is required which will ultimately affect the output file. Mismatched tape capacities can also affect the output file in that detection of physical end-of-tape (a reflective spot) during output causes the writing of an end-of-file record. The balance of the input file is then continued on another tape.

Input from Punched Cards. An output tape resulting from an input card file contains one tape record per card. Hollerith cards are translated to BCD and written as 80-character tape records with even parity. Binary cards (a 7 and 9 punch in column 1) are written in column binary representation as 160-character tape records with odd parity. Correspondence between card columns and tape records is as follows:



Upper = rows 12 through 3

Lower = rows 4 through 9

If the input file is a combination of binary and Hollerith cards, the resultant output tape contains a combination of binary and BCD records. The input end-of-file card causes the writing of an end-of-file record on the output tape.

B. Line Printer

General. Input files to be printed can contain binary or BCD records. Binary records are treated as BCD and the resultant printer output may or may not be intelligible. Printing is done with or without carriage control. If carriage control is elected, the first character of each record is interpreted as the control character and is not printed. The standard CONTROL DATA FORTRAN control character set is used.

Character	Action
0	Double space, print and single space
1	Eject page, print and single space
+	Print and suppress spacing
Any Other	Print and single space

PRINT tapes produced by programs written in 3200 FORTRAN, 3600 FORTRAN or FORTRAN '60 can be printed.

If carriage control is not elected, print spacing and page ejection is determined by channel 1 of the printer carriage control tape. Use of the standard tape results in single spacing with automatic ejection at end-of-page.

Input from Magnetic Tape. A tape record can contain one or several logical records. In the latter case, logical records must be separated by a record mark, a BCD character of value A82 (octal 32). Each logical record containing up to 120 characters is printed as a single line. Logical records containing more than 120 characters are printed as a series of 120-character lines.

If carriage control is used, the first character of each logical record is interpreted as the control character. For logical records longer than 120 characters, specified carriage control is applied only once. Excess characters are printed as one or more additional single spaced lines.

Input from Punched Cards. Each card is printed as a single line. If carriage control is elected, card column 1 is interpreted as the control character.

DATA TRANSMISSION

A principal ability provided by STEP is point-to-point data transmission. An input file read at one terminal is transferred via an inter-terminal data channel to the other terminal comprising the system. There it becomes an output file. Files are transferred in either direction but in only one direction at a time. As an option, transmission can involve terminal medium conversion. Standard input/output subprograms provide the following data transmission options:

1. Magnetic tape to remote magnetic tape
2. Magnetic tape to remote printer
3. Card to remote magnetic tape
4. Card to remote printer

A. Data Channel

The inter-terminal data channel consists of modulation/demodulation equipment (commonly called data sets) and a transmission facility. STEP operates with the following data channels:

Modulation Equipment:	DATA-PHONE Data Set 201A or 201B or equivalent
Transmission Facility:	2000- or 2400-BPS two- or four-wire private service, or 2000-BPS two-wire switched service

A common carrier (telephone or telegraph company) normally provides the data channel.

B. Data Flow

When engaged in data transmission both terminals operate together. A record is read, as a series of 6-bit characters, from the input file into processor storage at the Send terminal. The 6-bit characters are packed into 8-bit transmission words by STEP and a preamble of two start-of-message words, a count of the number of 8-bit transmission words within the packed record, and a record sequence number are created. The result is then presented to a data set controller one word at a time in a series of parallel transfers at a speed determined by data channel capacity. The controller adds a synchronization bit to each word and presents the 9-bit result to the data set as a serial bit stream. As the controller serializes a word, the processor is free to operate upon the word internally to update the record parity term. When the preamble and packed data have been transmitted the parity term is transmitted as four additional words.

At the Receive terminal the data set controller performs the reverse function of accepting a serial bit stream from the data set, stripping the synchronization bit and presenting an 8-bit word to the processor in a parallel transfer. STEP normally monitors the data channel looking for Start-of-Message words. When sensed, information following the second Start-of-Message word is input and stored within the

processor until the number of words prescribed by the preamble has been accumulated. Concurrent with input and storage, words are operated upon to develop a parity term. The regenerated term is compared with that transmitted by the Send terminal. If the terms are identical, the record is unpacked to form 6-bit characters written to the output file, and a reply indicating satisfactory reception and readiness to accept the next record is returned to the Send terminal.

If the terms are not identical, a negative reply indicating transmission error is returned immediately to the Send terminal and the error record is discarded. Negative reply or apparent lack of reply altogether (a possibility if momentary data channel failure occurs during transmission of the reply), causes the transmission to be repeated. The next input record is not processed until a positive reply is received for the current record. This process is repeated for each record in the input file and is called the basic record cycle (see figure 3).

C. Processing Overlap

Some operations always overlap. For example, parity generation is concurrent with transmission and parity regeneration is concurrent with reception. Conditioning the communication channel to carry data in the opposite direction from which it last carried data (commonly called channel turn around) is concurrent with conventional input/output. The degree to which other operations overlap depends primarily upon the characteristics of the input file. When processing an input file where the maximum length of any record within the file exceeds 500 characters conventional input, transmission/reception and conventional output are serial operations as previously described. If all of the records within the input file are 500 characters or less, conventional input overlaps with transmission and conventional output overlaps with reception. The extent of overlap has an effect on throughput capacity (see APPENDIX A).

D. Error Protection

An extremely important characteristic of the 8030 System is its ability to reliably detect record transformations within the inter-terminal data channel (transmission errors). The parity term developed and transmitted with each record is based upon a Bose-Chadhuri cyclic code having a Hamming distance of 6. Protection afforded by the parity term is as follows:

1. One through 5-bit transformations occurring anywhere within a record are always detected.

2. More than 5-bit transformations occurring within 27 contiguous bits of the record are always detected.

3. If more than 5-bit transformations occur within the record but not within 27 contiguous bits, the odds of detecting the transformation are better than 67 million to 1.

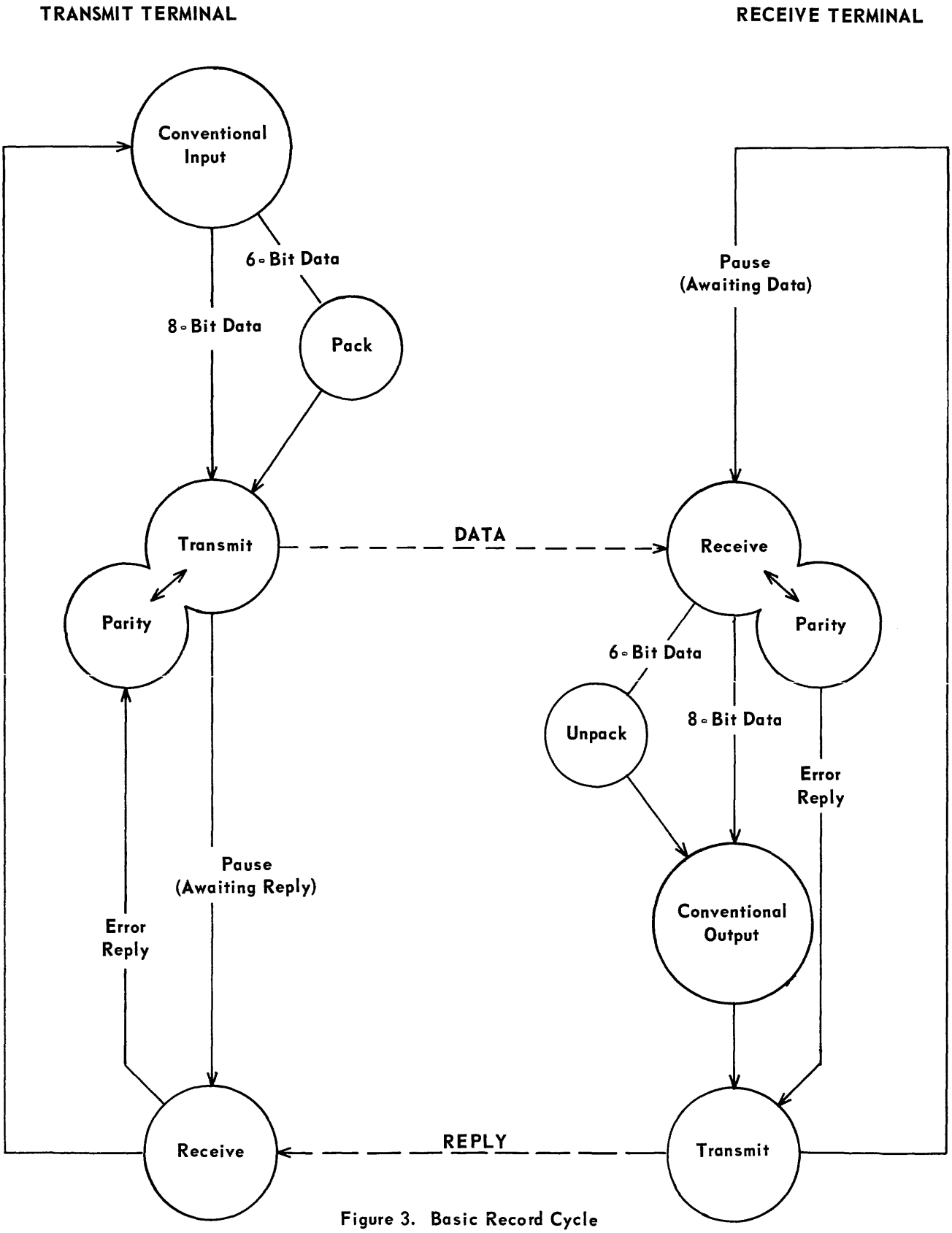


Figure 3. Basic Record Cycle

A second important characteristic is the ability of the 8030 to protect against lost records or duplicate records. The STEP-generated record sequence number is used for this purpose. Records arriving at the Receive terminal must bear consecutive numbers. An omitted number can only occur as a result of gross operator error and results in an immediate processing stop. Duplicate numbers can occur when a record is retransmitted because a positive reply from the Receive terminal is mutilated by data channel noise and not recognized at the Send terminal. Duplicate transmissions, consequently, are simply re-acknowledged and ignored.

LOCAL PERIPHERAL PROCESSING

When not engaged in data transmission, a terminal processor and its associated conventional peripherals can perform independent off-line peripheral processing operations.

One standard input subprogram and one standard output subprogram operate simultaneously to perform the off-line function. Given the hardware capabilities, any of the following medium conversion operations can be performed:

1. Card to magnetic tape
2. Card to printer
3. Magnetic tape to magnetic tape
4. Magnetic tape to printer

TERMINAL OPERATION

Each 8030 terminal has a record transmission control panel (see figure 4). This panel and the RUN and MANUAL INTERRUPT switches of the 8092-B provide a complete control facility for the entire terminal and its associated peripheral equipments. The panel provides the system operator with three basic functional abilities:

Selection

Control

Notification

A. Selection

There are two separate selection mechanisms on the panel:

Mode Selection

Peripheral Selection

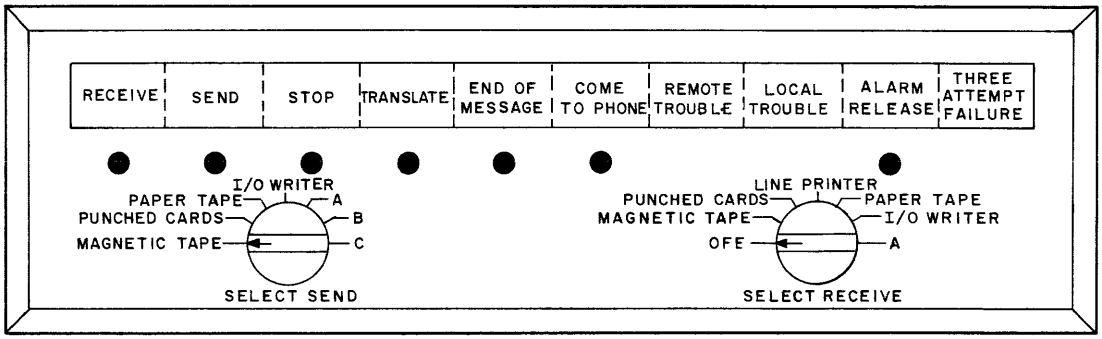


Figure 4. Record Transmission Control Panel

1. Mode Selection is accomplished using one of three available panel buttons:

   RECEIVE

   SEND

   TRANSLATE

   Pressing one of these buttons lights the corresponding indicator, and any other lit mode indicator extinguishes. Mode selection is performed at the start of each job and places the terminal in one of three modes for the remainder of the job:

   Receive Mode: The terminal operates as the receiving end of a two-terminal operation.

   Send Mode: The terminal operates as the sending end of a two-terminal operation.

   Translate Mode: The terminal operates as an independent local peripheral processing facility.

2. Peripheral selection is accomplished using two rotary switches on the lower portion of the panel:

   SELECT SEND

   SELECT RECEIVE

   These switches define the specific input and output peripheral equipment to be used for the job. Peripheral selection is performed at the start of each job for the entire job.

B. Control

The RUN switch, the MANUAL INTERRUPT switch, and the three panel buttons listed (along with their corresponding indicators) are used for job control:

STOP

END OF MESSAGE

ALARM RELEASE

1. STOP Button

   The STOP button temporarily stops a job in process. When a job is in process and the operator presses the STOP button and then the MANUAL INTERRUPT switch, the following events take place:

   a. All work in process makes an immediate orderly stop, but the job does not terminate.

   b. The STOP indicator and the ALARM RELEASE indicator light.

   c. The audible alarm sounds.

   Pressing the ALARM RELEASE indicator turns off the STOP and ALARM RELEASE indicators and, the audible alarm. Returning the 8092-B to the run condition restarts the job in process.

2. END OF MESSAGE Button

   Pressing the END OF MESSAGE button terminates a job in process. First the job is stopped, the END OF MESSAGE pressed and then the 8092-B RUN switch is returned to RUN. When the 8092-B switch is returned to RUN, the following events take place:

   a. The job in process immediately terminates and an END-OF-FILE is written on the output medium, if appropriate.

   b. The END OF MESSAGE indicator on each operating terminal lights.

   c. The audible alarm on each operating terminal sounds.

   Pressing the ALARM RELEASE button turns off the indicator and audible alarm.

   These end-of-message events also occur automatically when the 8030 detects an end-of-file condition on the input peripheral device.

3. ALARM RELEASE Indicator

The ALARM RELEASE indicator lights whenever the audible alarm sounds. The ALARM RELEASE button permits the operator to acknowledge any unusual system condition and to resume job operation following all operator controlled stops and pauses. When the operator presses the ALARM RELEASE button, any of the following indicators which are on will extinguish:

STOP

END OF MESSAGE

REMOTE TROUBLE

LOCAL TROUBLE

ALARM RELEASE (and the audible alarm)

C. Notification

Two indicators notify the terminal operator of specific unusual conditions:

REMOTE TROUBLE

LOCAL TROUBLE

When these indicators light, the audible alarm sounds. When the condition is corrected and the ALARM RELEASE button pressed, these indicators and the audible alarm go off.

D. Job Processing

The record transmission control panel permits jobs to be processed with a minimum of operator action and attention. Once the job begins, conditions which require operator intervention are indicated visually and by an audible alarm. The operator can thus attend to other functions without sacrificing transmission efficiency or job speed. To start any job requires only the following steps.

1. Set up the peripheral device.
2. Establish the phone connection, if necessary.
3. Select the terminal mode.
4. Select the input and output peripheral device.
5. Stop.
6. Press the ALARM RELEASE button.
7. Run.

The job begins and terminates normally with no further intervention when an end-of-file condition is detected on the input peripheral device.

APPENDIX A

THROUGHPUT CALCULATION

A useful criterion for evaluating the performance capability of a point-to-point data transmission system is effective transmission rate. This is the average rate, in 6-bit characters per second, at which an input file can be moved through the inter-terminal data channel.

Six principal factors influence this rate:

1. Data channel operating bit rate
2. Time to reverse the direction of data flow within the channel (commonly called turn-around time.)
3. Frequency of turn-around
4. The redundancy of the transmission scheme (i.e., the proportion of bits transmitted which are used for error protection or synchronization and do not actually represent information from the input file)
5. The degree to which the data channel is idle due to terminal processing
6. Data channel error rate

The first five of these factors are quantitatively related in the family of formulas developed in the following pages. Use of the appropriate formula allows prediction of the approximate theoretical effective transmission rate for a given set of conditions (input medium, mean input record length, data channel speed, data channel turn around time, output medium).

The effect of data channel errors is extremely difficult to determine quantitatively because they tend to occur in bursts and vary in quantity depending upon the quality of the data channel. To some extent, errors reduce the calculated effective transmission rate because of the retransmission necessary for recovery.

The terms "external record" and "internal record" which appear in the formulas are defined in detail in APPENDIX B. In general terms, an external record is the unit increment of data read from or written to a peripheral, and an internal record is the representation of one or more external records in processor storage.

FORMULAS

These formulas apply to 6-bit data. If the internal record prepared by an input subprogram contains 8-bit data, the underlined terms in the formulas should be deleted. C and N will then be in terms of 8-bit characters.

Let:

C = effective transmission rate in 6-bit characters/second

N = number of 6-bit characters in internal record assembled by the input subprogram

R = time, in ms, to read external record(s) making up an internal record

P_r = time, in ms, to do special post-read processing of internal record

P_w = time, in ms, to do special pre-write processing of internal record

W = time, in ms, to write internal record as external record(s)

t = data channel turn-around time, in ms:

160 for two-wire service (nominal value)
16 for four-wire service (nominal value)

K = data channel time per character, in ms:

3.75 for 2400 BPS data channel
4.50 for 2000 BPS data channel

And,

$T_r = t - (R + P_r + .28N)$

$T_w = t - (W + P_w + .28N)$

$t_r = (R + 2P_r + .6N) - (K(.75N + 19) + 2t)$

$t_w = (W + 2P_w + .6N) - (K(.75N + 19) + 2t)$

T = The larger of t_r and t_w

Then, for Class I files (those containing any record > 500 characters in length)

$$C = \frac{1000N}{R + P_r + W + P_w + K(.75N + 19) + .56N + T_r + T_w} \qquad (1)$$

where values of T_r and T_w are ignored if ≤ 0

For Class II files (those where all records are ≤ 500 characters in length)

$$C = \frac{1000N}{K\ (.75N + 19) + 2t} \qquad (2)$$

for values of $T \leq 0$, and

$$C = \frac{1000N}{K\ (.75N + 19) + 2t + T} \qquad (3)$$

for values of $T > 0$

EXAMPLE 1.

Consider the case where 1000-character magnetic tape records are being transmitted over a 2400-BPS two-wire data channel using standard 601 magnetic tape input/output subprograms.

Then,

$N = 1000$

$R = 63$ ms

$P_r = 1$

$P_w = 1$

$W = 63$ ms

$t = 160$ ms

$K = 3.75$

$T_r = 160 - (63 + 1 + .28\ (1000))$

$= 160 - 344$, or $- 184$

$T_w = 160 - (63 + 1 + .28\ (1000))$

$= 160 - 344$, or $- 184$

Applying (1)

$$C = \frac{1000 \cdot 1000}{63 + 1 + 63 + 1 + 3.75\,(.75(1000) + 19) + .56(1000)}$$

$$= \frac{1000 \cdot 1000}{128 + 2884 + 560}$$

$$= \frac{1{,}000{,}000}{3572}$$

$$= 280 \text{ characters/second}$$

EXAMPLE 2.

If 120-character records were being transmitted over a 2400-BPS four-wire data channel using the standard 601 magnetic tape input subprogram and the standard 166-2 line printer output subprogram,

then,

$N = 120$

$R = 21$

$P_r = 1$

$P_w = 1$

$W = 422$

$t = 16$

$k = 3.75$

$t_r = (21 + 2 + 72) - (3.75(90 + 19) + 32)$

$= 95 - 409$ or -314

$t_w = (422 + 2 + 72) - (3.75(90 + 19) + 32)$

$= 496 - 409$ or 87

$T = 87$

Applying (3)

$$C = \frac{1000 \cdot 120}{3.75\,(90 + 19) + 32 + 87}$$

$$= \frac{120{,}000}{528}$$

$$= 227 \text{ characters/second}$$

NOTE

The limiting factor in this operation is the ability of the receive processor to restore 8-bit words to 6-bit data conjuncted with the speed of the printer (i.e., the data channel is at times idle when printing is taking place). The calculated effective rate is probably somewhat lower than the actual rate because the term t_w is a conservative approximation which does not account for the overlap of data restoration with output of the preceeding record. That portion of figure 5 which shows the relationship between processor operations when transmitting a CLASS II FILE illustrates this anomaly.

TIMING VALUES FOR STANDARD I/O SUBPROGRAMS

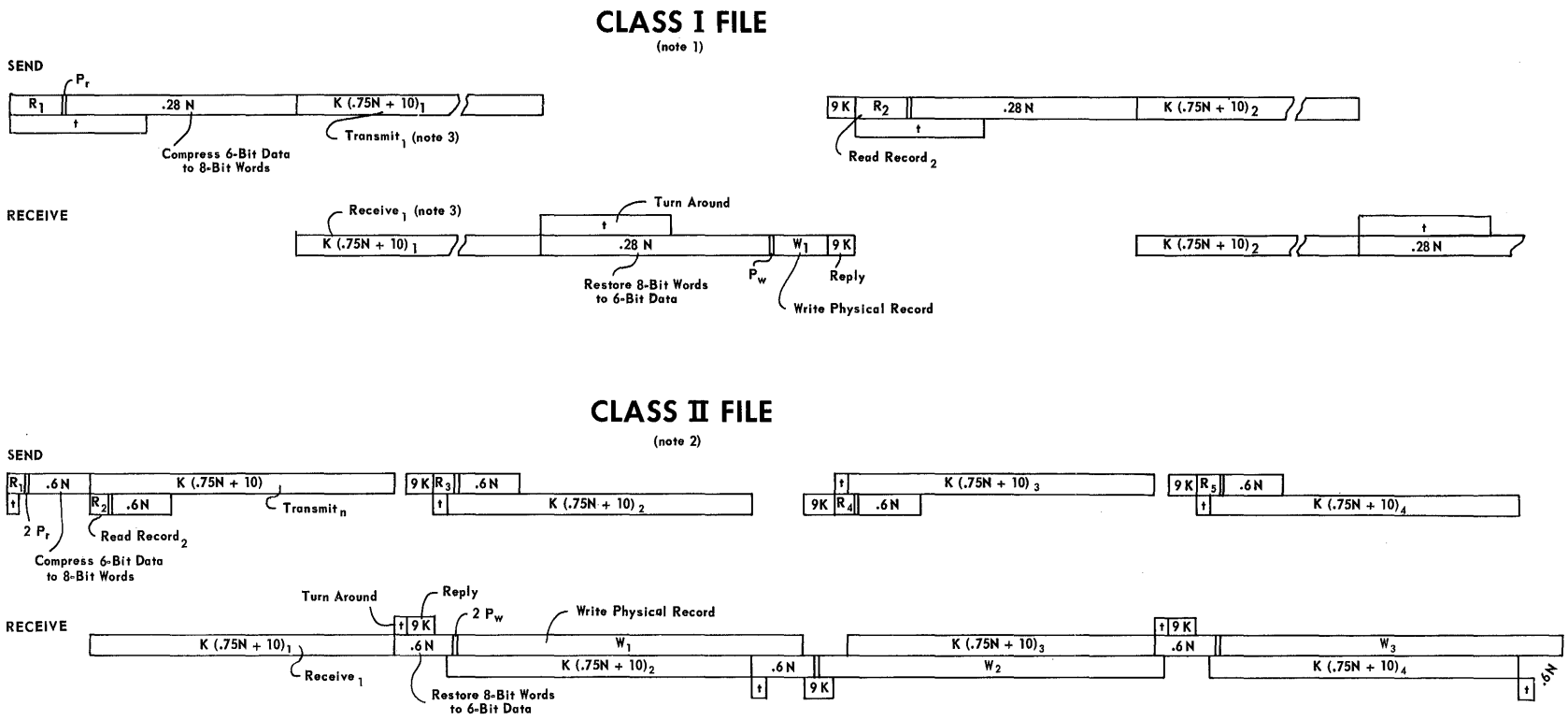
601 Magnetic Tape	200 Char/In.	556 Char/In.
R or W = 15+	0.133 ms/char	0.048 ms/char
P_r = 1 ms		
P_w = 1 ms		

405 Card Input	51 Columns	80 Columns
R = 21.7 +	20 ms	31 ms
P_r = 1 ms		

166-2 Printer Output

W = 422 ms

P_w = 1 ms



NOTES:

1. Illustration of **EXAMPLE 1.**
2. Illustration of **EXAMPLE 2.**
3. Foreshortened. Actual Time = 2850 ms

Scale: 1 ⊢⊣ = 25 ms

Figure 5. Timing Relationships Between Processor Operations During Record Transmission

APPENDIX B

STEP ALTERATION

A. Appendix Scope

Conventional input/output subprograms included with STEP obviously do not satisfy all conceivable needs. To provide open-ended flexibility, STEP has been structured to permit easy incorporation of user implemented conventional I/O subprograms.

The balance of this appendix describes the environment provided by the STEP nucleus within which an input or output subprogram must operate, and sets forth the rules and techniques for implementing and incorporating one. Reader familiarity with the programming characteristics of the CONTROL DATA 8092-B TeleProgrammer is assumed.

B. Basic Role of I/O Subprograms

Two definitions are pertinent to subsequent discussion.

External Record. An external record is the unit increment of data read from or written to a peripheral medium. One magnetic tape block, one punched card, and one print line are examples of external records.

Internal Record. An internal record is a data construct within the core storage of the 8092-B formed from one or more external records or parts thereof, internally generated data, or a combination of both external and generated data.

The basic role of an input subprogram is to prepare an internal record. Except for overall size limitation, the content and format of an internal record are at the discretion of the input subprogram. For example, a subprogram could be written to translate an external record to a different code, or to pack several punched cards into a single internal record.

The basic role of an output subprogram is to dispose of an internal record. Disposition is at the discretion of the subprogram. For example, an internal record could be discarded, written to an output medium as a single external record, as several external records, or written to more than one output medium.

C. Layout of 8092-B Storage

8092-B storage is divided into three major functional areas as shown in figure 6.

Boundaries between these areas are fixed and cannot be altered without major STEP modification.

Area I. Area I is where an internal record is assembled by an input subprogram and where an internal record is passed to an output subprogram for disposition.

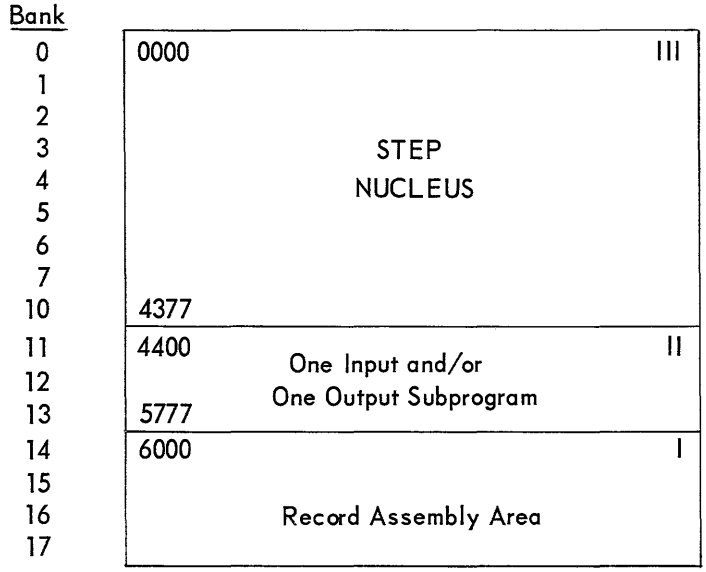


Figure 6. 8092-B Storage Layout

In the case of input, Area I can be used as an input buffer directly (this is done with standard STEP input subprograms), or actual input can be into Area II and data then moved under program control to Area I. The latter approach might be convenient if one were editing several cards into a single internal record and wished to form an internal record of maximum possible size (a factor in effective transmission rate.)

In the case of output, the same options exist. Data can be output directly from Area I, or moved to Area II under program control and output from there. The choice of technique is strictly a matter of usefulness to the implementer in either case.

Area II. This area is primarily for I/O coding, although it can be used for input/output of external records as previously described. To plan an input or output subprogram, some insight into STEP operational philosophy is essential. STEP is supplied to the user as a library. Job initiation causes the STEP nucleus to be loaded along with a transient panel interpretation program which operates in Area I. The panel interpretation program loads Area II according to the settings of the operator's panel.

	TERMINAL MODE		
	SEND	RECEIVE	TRANSLATE
SELECT SEND Switch			
– Mag Tape	5		5
– Punched Cards	6		6
– Paper Tape	7	switch ignored	7
– I/O Writer	10		10
– A	11		11
– B	12		12
– C	13		13
SELECT RECEIVE Switch			
– Off		14	14
– Mag Tape		15	15
– Punched Cards	switch ignored	16	16
– Line Printer		17	17
– Paper Tape		20	20
– I/O Writer		21	21
– A		22	22

The numbers represent file numbers of programs in the STEP library. Files 5 through 13 are input subprograms, and files 14 through 22 are output subprograms. If, for example, terminal mode is Send and the SELECT SEND switch is set to MAGNETIC TAPE, file 5 from the library is loaded to Area II. The SELECT RECEIVE switch is ignored.

However, if terminal mode is Translate, both the SELECT SEND and SELECT RECEIVE switches are interpreted and two files are loaded to Area II.

STEP library programs are in absolute machine language. The placement of a program within Area II at the time it is loaded, is predetermined by assignment of an origin address at the time the program is assembled. This means that input and output subprograms which will simultaneously occupy Area II during a Translate (local peripheral processing) operation must be planned to avoid storage conflict.

In summary:

- An input subprogram used only in the Send mode can occupy 1377_8 storage locations.
- An output subprogram used only in the Receive mode can occupy 1377_8 storage locations.
- An input/output subprogram pair used in Translate mode must share 1377_8 storage locations and must be pre-planned so as not to conflict.

Area III. Primarily this area is used for STEP coding, and is of concern to one implementing an I/O subprogram only because it contains the tables used to communicate parameters. These tables are all located in bank 0 because of its unique addressing properties.

D. Ground Rules for Implementation

Both input and output subprograms are subject to the following ground rules:

1. Data input/output must be performed on the 8092-B buffer channel.
2. Peripherals must be programmed so as not to generate external interrupts.
3. Interrupt must not be inhibited for more than 200 usec at any one time.

E. Nucleus/Subprogram Communication

Communication between the STEP nucleus and an input or output subprogram is accomplished via tables identified by symbolic labels. When implementing a subprogram, table references should be coded in terms of these labels. Label values can be assigned either by using EQUALS statements or by using the STEP symbol table at time of assembly.

1. Control Transfers

   Entry to I/O Subprograms. One of the functions of the STEP nucleus is to enter an input subprogram whenever an internal record is needed and to enter an output subprogram whenever an internal record must be disposed of.

When an input subprogram (IS) or an output subprogram (OS) is initially loaded, STEP establishes its first word address (FWA) as an entry. This address consists of a bank tag (four bits) and a local address within the bank (eight bits). These values are located in Table 17.

Label	Content	Form
TBL17+0	IS Local Address	xxxxxxxx
TBL17+1	IS Bank Tag	ooooxxxx
TBL17+2	OS Local Address	xxxxxxxx
TBL17+3	OS Bank Tag	ooooxxxx

Control is always transferred to an I/O subprogram at the address shown in Table 17. The entry sequence used by the nucleus results in tag register 1 being set to the bank tag, when control appears at the subprogram.

A subprogram can change Table 17 at will, in effect using it as a switch during operation. At end-of-job STEP restores TBL17 to its initial value (subprogram FWA).

Not Complete Exit. When assembling or disposing of an internal record, I/O subprograms should return control to the nucleus whenever they are in a wait condition (after a buffer has been initiated for example). This allows the nucleus executive to seek other system work. If no work is found, or when other work has been completed, control is returned to the subprogram at the entry currently defined by Table 17. If the subprogram still cannot profitably use the processor, it should again exit. The not-complete exit sequence is:

```
        ◆              Comment
        ◆
LDN     ◆              Bank tag of nucleus re-entry to A
        WORK
ATT     T1             Set tag register
UJP     T1             Exit from output subprogram
        WORK1

        or

UJP     T1             Exit from input subprogram
        WORK2
```

Operation Complete Exit. When an I/O subprogram has completed assembly or disposal of an internal record, it notifies the nucleus by making a unique operation complete exit. The nucleus then makes the exiting subprogram inactive until its services are next required. The exit sequence is:

```
        ◆                   Comment
        ◆
LDN     ◆                   Bank tag of nucleus reentry to A
          WORK
ATT       T1                Set tag register
UJP       T1                Exit from output subprogram
          WORK21

        or

UJP       T1                Exit from input subprogram
          WORK22
```

2. Parameters Related to Normal Operation

Applicable to Both Input and Output Subprograms.

Label	Content
TBL17+6	Buffer channel status word. If (TBL17+6) = 0, the channel and tag register 3 are available. (TBL17+6) > 0, the channel and tag register 3 are in use for input (TBL17+6) < 0, the channel and tag register 3 are in use for output This word is set and cleared by I/O subprograms.

Applicable to Input Subprograms at Entry.

Label	Content	Form
TBL3+1	Current assembly area first word bank tag. The local address is implicit 0.	ooooxxxx
TBL3+2	Current assembly area last work bank tag.	ooooxxxx
TBL3+3	Current assembly area last word local address.	xxxxxxxx

These values define the area in which the input subprogram is to assemble an internal record. For convenience, call the 12-bit address formed from the first word bank tag and the implicit local address 0, START. (START is a calculated address and must not be used as a symbolic label).

Expected from Input Subprogram at Operation Complete Exit.

Location	Content	Form
START+1	Bank tag of last bank within the assembly area containing data.	ooooxxxx
START+2	Local address +1 of the last word containing data.	xxxxxxxx
START+3	Special status bits.	

Bit	Use
7	Set to 1 to indicate input end-of-file, otherwise 0.
6	Not used
5	Not used
4	Set to 1 if data is 6-bit data
3	Not used
2	Set to 1 if input is* from magnetic tape
1	Not used
0	Set to 1 if data is* odd parity, 0 if data is even parity.

START+4	Available for own assignment. This word, along with the three above is communicated to the output subprogram.	
START+5	Data	
◆		
◆		
◆		
◆		
◆		

*These bits are required only if output is to be performed by the standard STEP magnetic tape output subprogram.

Applicable to Output Subprograms at Entry.

Label	Content	Form
TBL4+1	Current internal record first word bank tag. The local address is implicit 0.	ooooxxxx
TBL4+2	Current internal record last word bank tag.	ooooxxxx
TBL4+3	Current internal record last word local address.	xxxxxxxx

These values define the assembly area which contains the internal record to be disposed. For convenience, call the 12-bit address formed from the first word bank tag and the implicit local address 0, START'. (START', like START, is a calculated address and must not be used as a symbolic label.) Then, for internal record $_n$

$$(START' + 1)_n \equiv (START + 1)_n$$

$$(START' + 2)_n \equiv (START + 2)_n$$

$$(START' + 3)_n = (START + 3)_n$$

$$(START' + 4)_n = (START + 4)_n$$

where $\equiv$ is to be interpreted in the logical sense as equivalent to, and $=$ has the conventional meaning equal to.

As a by-product of packing/unpacking operations when transmitting 6-bit data, the relative position of the data within the internal record is changed. (START' + 1) and (START' + 2) are adjusted to compensate for this movement. Two additional parameters giving current data first word address must be accessed to locate the data.

Label	Content	Form
TBL17+7	Data first word local address.	xxxxxxxx
TBL17+10	Data tag increment. The sum of (TBL17+10) and(TBL4+1) yields data first word bank tag.	ooooxxxx

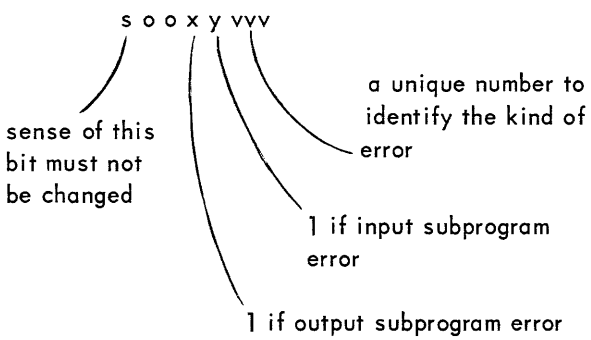
Expected From Output Subprogram at Operation Complete Exit. If the input end-of-file bit is set in START'+3 at entry, no more records are forthcoming and the output subprogram should execute its end-of-file sequence before operation complete exit is made. As part of the end-of-file sequence the following parameters should be stored:

Label	Content
TBL18 +3	Set to 001
TBL18 +4	Set to 120
TBL18 +13	Set to 120

End-of-file is the only non-error condition under which exit parameters are required from an output subprogram.

3. Parameters Related to 1/0 Errors

If a non-recoverable input or output error occurs, the subprogram should supply the following parameters and make a not-complete exit:

Label	Content
TBL17+4	
TBL18+3	Set to 200

s o o x y vvv
sense of this bit must not be changed
a unique number to identify the kind of error
1 if input subprogram error
1 if output subprogram error

The use of bits "vvv" is a convention followed in standard I/0 subprograms; they can be used in another way if the user desires. STEP stops with the content of the lower six bits of this location displayed in A.

STEP permits the operator to enter a value in A when an error Stop occurs. When the operator returns the processor to Run, the value he has loaded in A is stored to
TBL17+5
and control returned to the subprogram at the current entry address. If the operator decides to terminate the job, the value of TBL17+5 will be 200 and the subprogram must execute its end-of-file sequence

Other values can be interpreted by the subprogram as unique action requests. Standard I/0 subprograms interpret the operator supplied value according to the following conventions:

(TBL17+5) = 0, Try operation again

(TBL17+5) > 0, Accept record as read (or let output in error stand) and continue.

F. Subprogram Working Storage

24_8 storage locations in bank 0 are reserved for I/0 subprograms working store. These locations are divided into two 12_8 word tables. Table 14, which has a base label TBL14, is intended for input subprogram use and Table 15, which has a base label TBL15, is intended for output subprogram use. Both of these tables are initialized to zero by STEP at start-of-job.

G. Subprogram Logical Organization

Figures 7 and 8 show the logic of a simple tape input subprogram and a printer output subprogram respectively. These figures illustrate the use of the not-complete and operation-complete exits and entry/exit parameters. In actual practice, other error checking logic would be incorporated to prevent an infinite loop if a peripheral failed by becoming permanently busy. In the case of tape input, automatic re-read attempts would be incorporated and the operator would not intervene until several had been tried.

The input subprogram shown assumes that one external tape record becomes one internal record. Likewise, the output subprogram assumes that one internal record becomes one print line.

H. Subprogram Preparation

Subprograms should be coded in TOSAS, an assembly language for the 8092 TeleProgrammer described in the 8092 TeleProgrammer Programming Reference Manual (pub. no. 368 107 00). TOSAS operates on the 8090 or 160-A Computer.

The only programming rule, aside from those already developed regarding entry, exit, tag registers and parameters, is that programs must begin with an ORG statement which specifies subprogram FWA (and implicitly initial entry) and end with an END statement.

I. STEP Library

The STEP library is a basic library which can be updated according to the rules set forth in the manual, A Basic Library System For The TeleProgrammer (pub. no. 368 235 00).

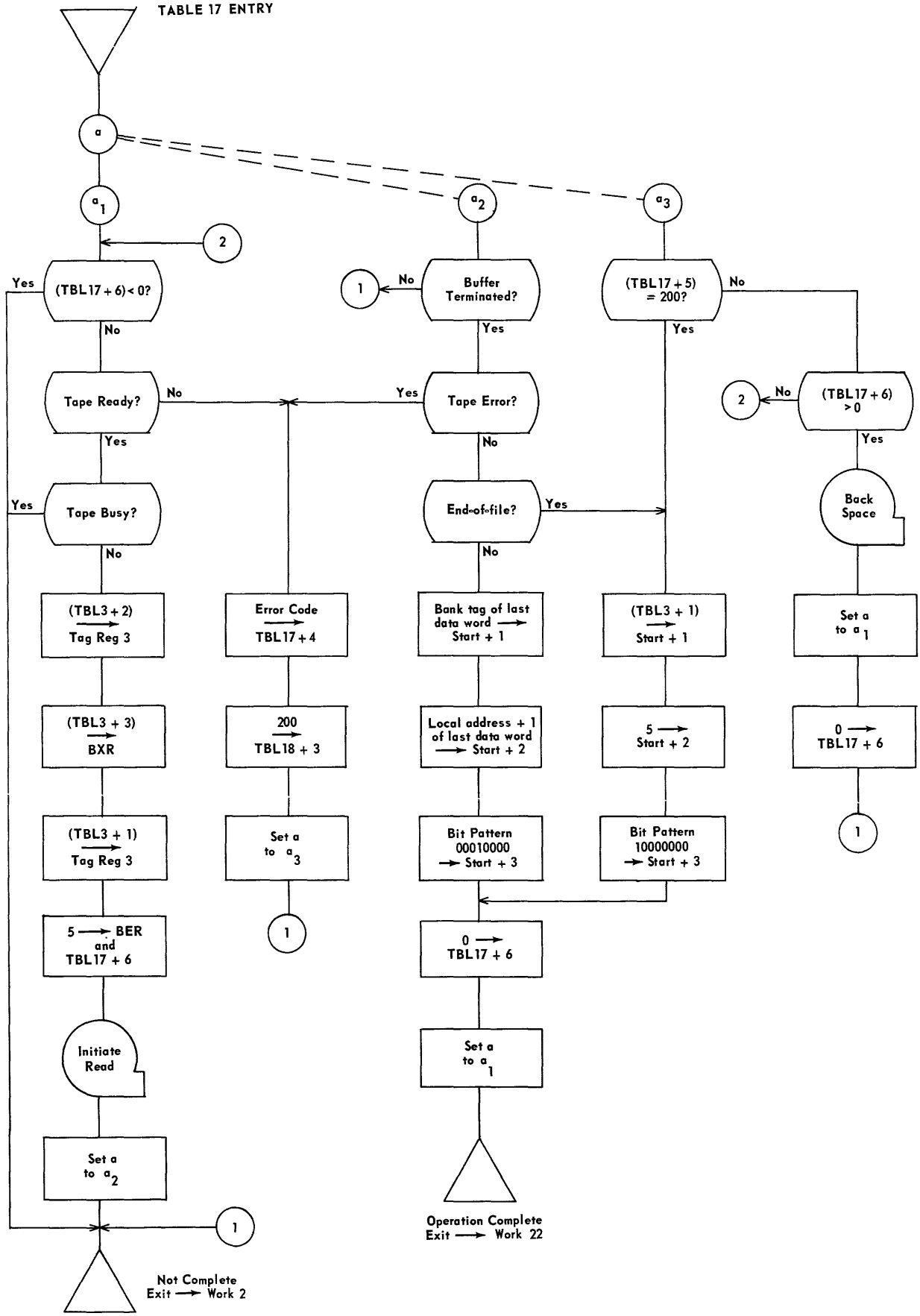
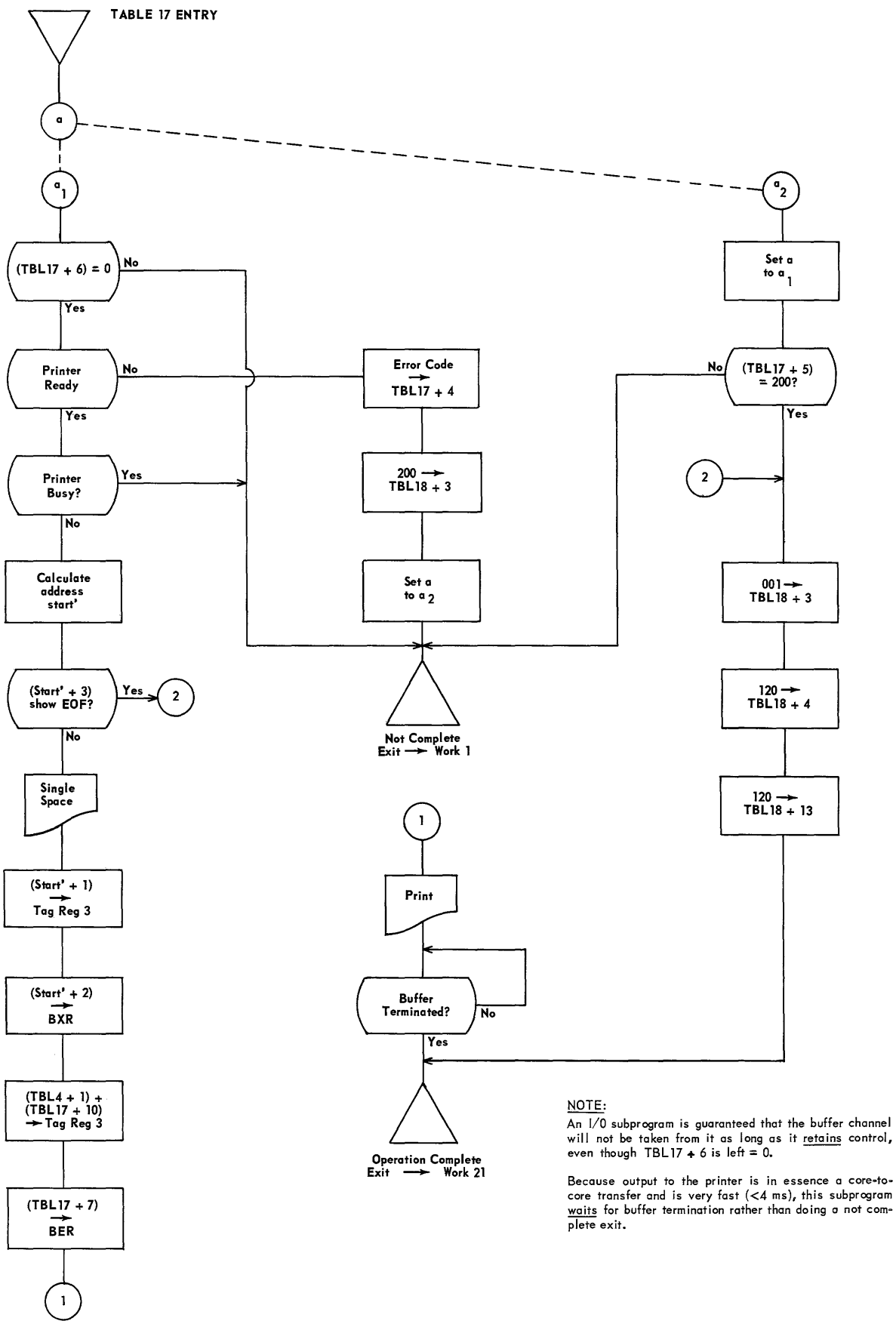


Figure 7. Sample Input Subprogram



NOTE:
An I/O subprogram is guaranteed that the buffer channel will not be taken from it as long as it retains control, even though TBL17 + 6 is left = 0.

Because output to the printer is in essence a core-to-core transfer and is very fast (<4 ms), this subprogram waits for buffer termination rather than doing a not complete exit.

Figure 8. Sample Output Subprogram

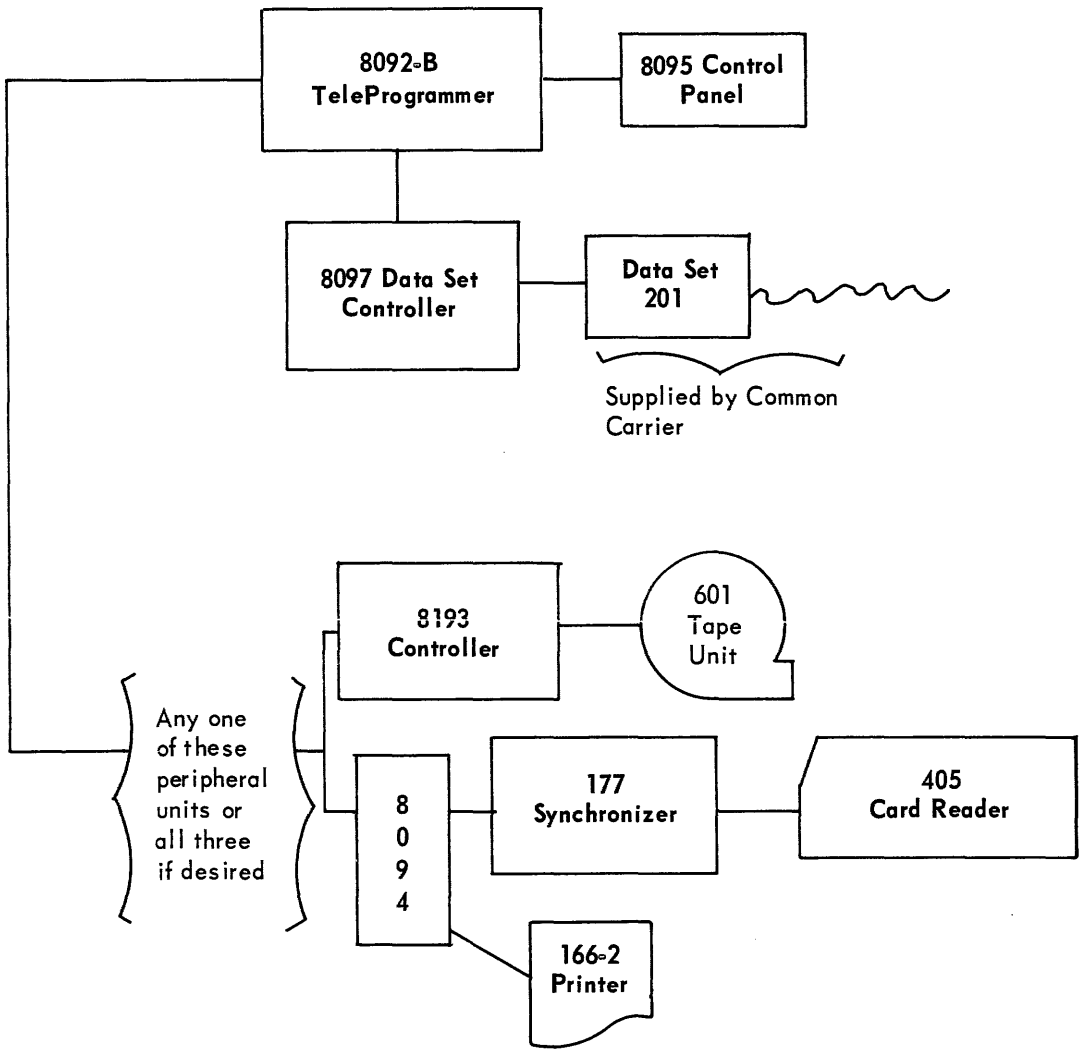
APPENDIX C

BCD CODES

BCD Code	Character	Hollerith Code	BCD Code	Character	Hollerith Code
00	:	illegal	40	-	11
01	1	1	41	J	11 1
02	2	2	42	K	11 2
03	3	3	43	L	11 3
04	4	4	44	M	11 4
05	5	5	45	N	11 5
06	6	6	46	O	11 6
07	7	7	47	P	11 7
10	8	8	50	Q	11 8
11	9	9	51	R	11 9
12	0	0	52	%	11 0
13	=	3 8	53	$	11 3 8
14	≠	4 8	54	*	11 4 8
15	≤	5 8	55	↑	11 5 8
16	'	6 8	56	↓	11 6 8
17	[	7 8	57	>	11 7 8
20	blank	unpunched	60	+	12
21	/	0 1	61	A	12 1
22	S	0 2	62	B	12 2
23	T	0 3	63	C	12 3
24	U	0 4	64	D	12 4
25	V	0 5	65	E	12 5
26	W	0 6	66	F	12 6
27	X	0 7	67	G	12 7
30	Y	0 8	70	H	12 8
31	Z	0 9	71	I	12 9
32	]	0 2 8	72	<	12 0
33	,	0 3 8	73	.	12 3 8
34	(	0 4 8	74	)	12 4 8
35	→	0 5 8	75	≥	12 5 8
36	≡	0 6 8	76	?	12 6 8
37	~	0 7 8	77	;	12 7 8

APPENDIX D

MINIMUM 8030 TERMINAL CONFIGURATION



NOTE:
Either a magnetic tape unit or a paper tape reader must be supplied for program loading.